Охота за аномалиями на графиках

Александр Барановский badoo





Привет, меня зовут

Александр Барановский

https://github.com/badoo
tech.badoo.com
@efisuby



Зачем?

Зачем мы это делаем

Профит для бизнеса

- Эффекты маркетинговых компаний
- Контроль за бизнес-метриками

Профит для инженеров

- Отслеживание технических метрик
- Помощь при исследованиях

О чем пойдет речь?

Погружаемся в теорию

Определяем тулсет

Делаем подготовку

Готовим модели предсказаний

Строим доверительный интервал

Определяем лучшую модель

Сохраняем результаты

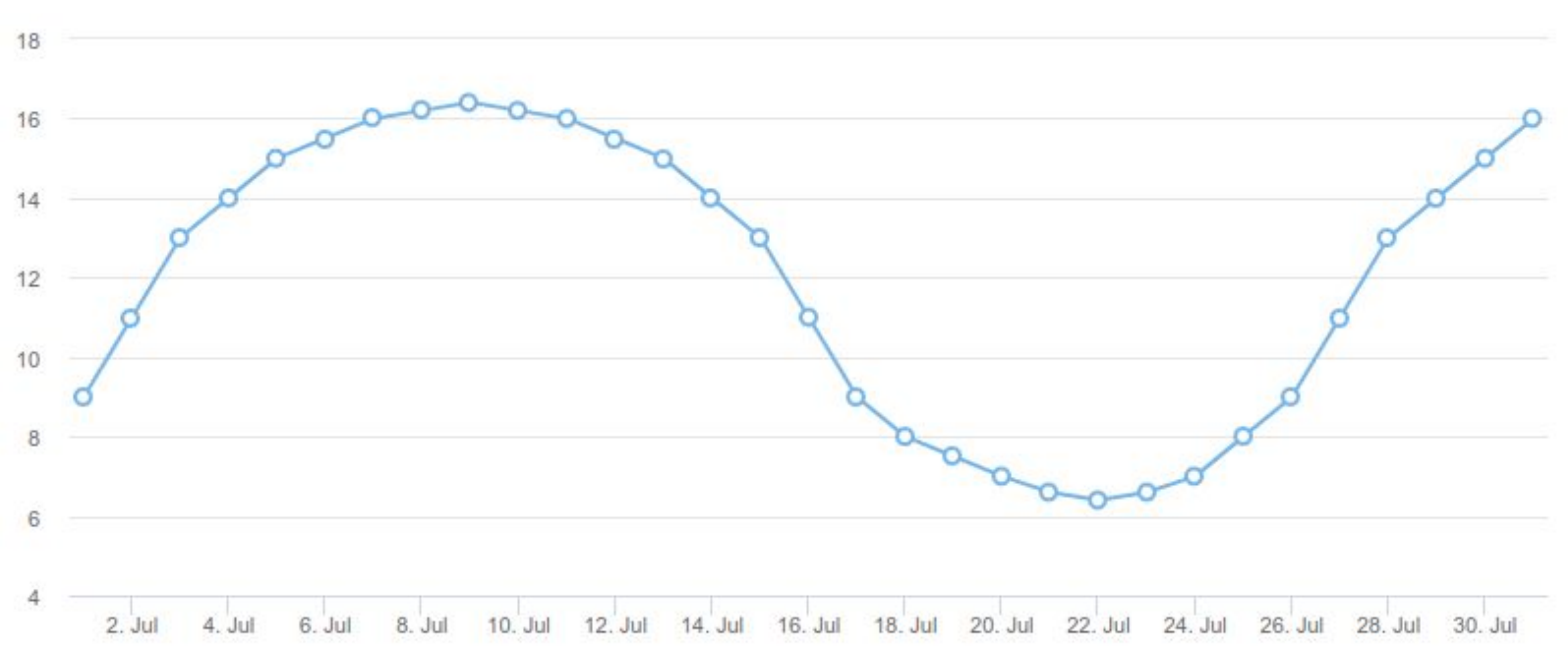
Учимся работать с множеством аномалий

Подводим итоги

Немного теории

Теория:

Графики



Теория:

Dimensions

Dimension (срез) — набор характеристик, определяющих природу данных графика

Теория:

Dimensions

Примеры дименшенов:

- Страна
- Устройство пользователя
- Оператор
- ...

Теория:

Dimensions

Проблема комбинаторного взрыва

- 100 стран
- 50 разных устройств
- 200 операторов

Всего 1.000.000 графиков, и **это не предел**

Теория:

Аномалия

Аномалия — это отклонение от нормы, от общей закономерности

Теория:

Аномалия



Теория:

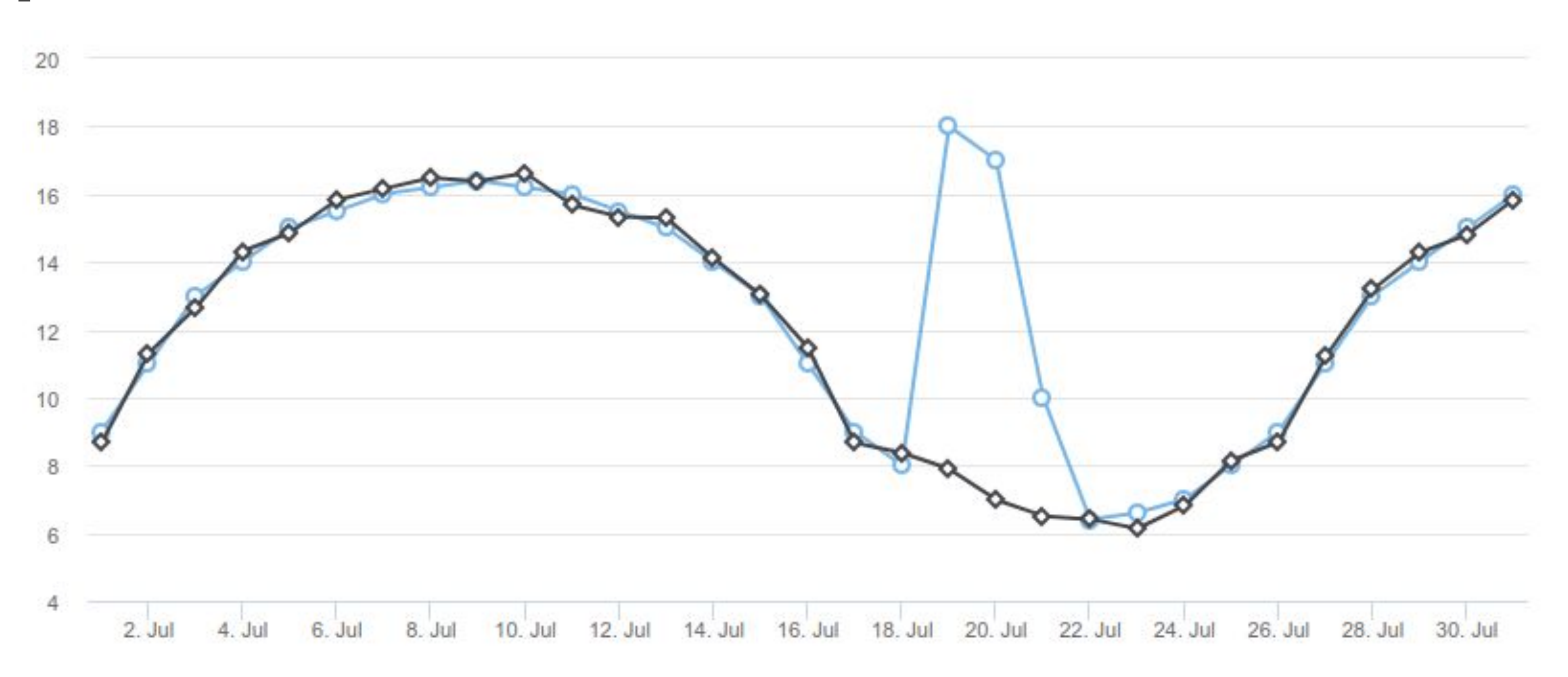
Предсказание

Предсказание (forecast, prediction) — это по определению сообщение о некотором событии, которое непременно произойдет в будущем

One step ahead forecast — предсказание на один шаг вперед

Теория:

Предсказание



Теория:

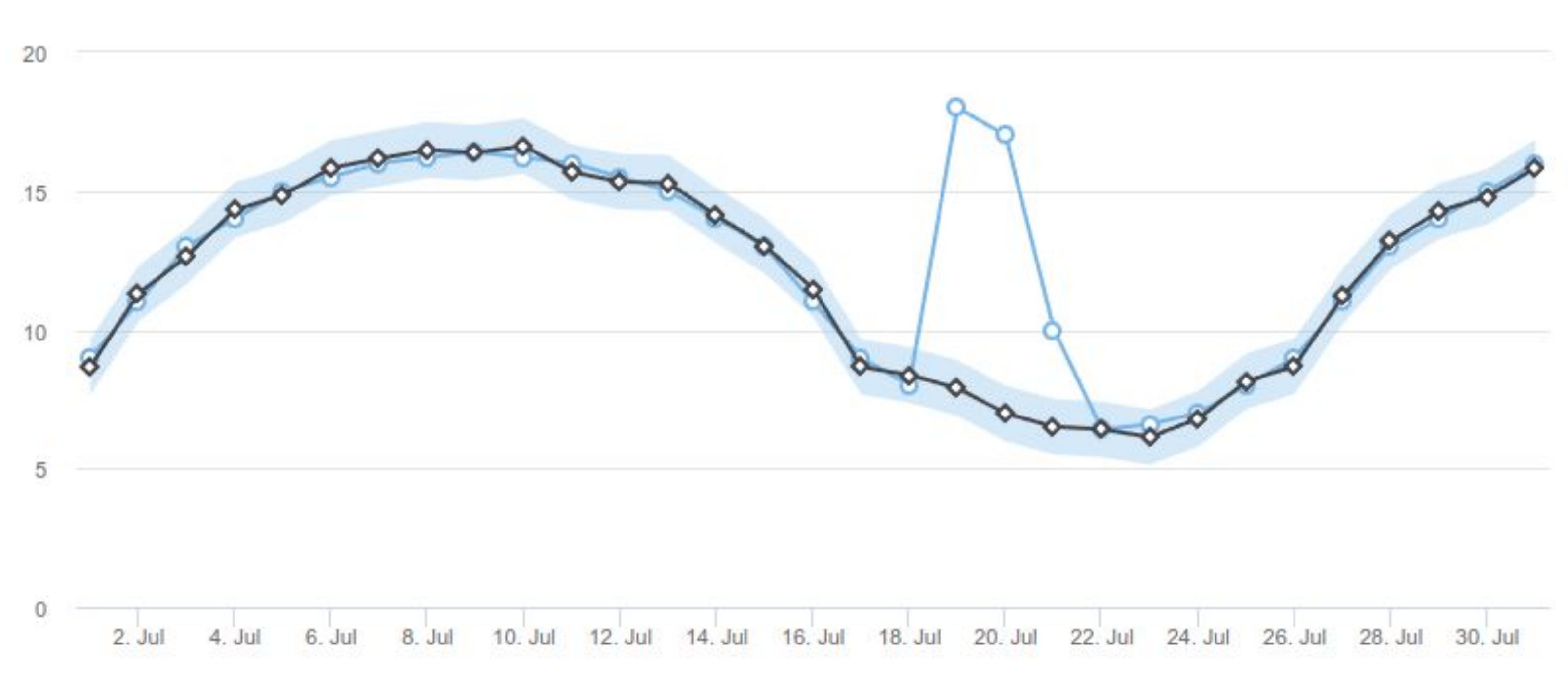
Доверительный интервал

Доверительным называют интервал, который покрывает неизвестный параметр с заданной надежностью

Доверительный интервал — на сколько можно ошибиться в предсказании

Теория:

Доверительный интервал



Тулинг

Тулинг:

Базы данных

Exasol

Vertica

Snowflake

Oracle

BigQuery

Clickhouse

...

Подготовка

Схема хранения:

Получение данных

```
CREATE TABLE data_source (
   id Int64,
   dimension_1 String,
   dimension_2 String,
   metric String,
   ts DateTime,
   value Int64
) ENGINE=MergeTree PARTITION BY (ts)
ORDER BY (ts, id)
```

- Есть ID
- Список дименшенов
- Описание измерения
- Временная отметка
- Значение

Таблица с оригинальными данными

Схема хранения:

Подготовка: метаданные

```
CREATE TABLE dimensions (
   id Int64,
   dimension_1 String,
   dimension_2 String,
   metric String,
   first_ts AggregateFunction(min, DateTime),
   last_ts AggregateFunction(max, DateTime)
) ENGINE=AggregatingMergeTree()
PARTITION BY ID % 32 ORDER BY id;
```

```
insert into dimensions
select id, dimension_1, dimension_2, metric, ts, ts
from data_source;
```

Выносим все метаданные

Получаем временные метки первого и последнего появления данных

Заполняем каждый раз при анализе нового периода

Схема хранения:

Подготовка: значения графиков

```
CREATE TABLE values_and_predictions (
   id Int64,
   ts DateTime,
   value Float64,
   prediction_1 Float64,
   threshold_1 Float64,
   prediction_2 Float64,
   threshold_2 Float64,
   prediction_N Float64,
   threshold_N Float64,
   flags UInt64
) ENGINE=SummingMergeTree([id, ts])
PARTITION BY (ts) ORDER BY (ts, id)
```

Из метаданных только ID

Отдельные поля для предсказаний и их доверительных интервалов

Возможность быстро проверить модели предсказаний, используя битовые маски в fields

```
insert into values_and_predictions (id, ts, value, flags)
select id, ts, value, 1 from data_source where ts = '2021-01-01 00:00:00'
```

Схема хранения:

Подготовка: предсказания

```
insert into values_and_predictions (id, ts, prediction_%NUM%, flags)
select id, '2021-01-01 00:00:00', prediction, bitShiftLeft(1, %NUM%)
FROM (
   %MODEL SELECT QUERY%
)
WHERE isFinite(prediction);
```

1. NUM — номер модели предсказаний
2. MODEL SELECT QUERY — запрос на получение данных предсказаний. Должен возвращать id и prediction

Схема хранения:

Подготовка: История значений

```
create table history (
   id Int64,
   ts DateTime,
   dimension_1 String,
   dimension_2 String,
   metric String,
   actual_value Float64,
   predicted_value Float64,
   threshold Float64
) ENGINE=MergeTree() PARTITION BY ts ORDER
BY ts, id;
```

Результаты предсказаний

Данные не удаляются

Подготовка завершена

1. Данные лежат и готовы для анализа
2. Метаданные не мешают
3. Руки чешутся предсказывать

Модели предсказаний

Предсказания:

Наивная

$Y(t+1)=Y(t)$

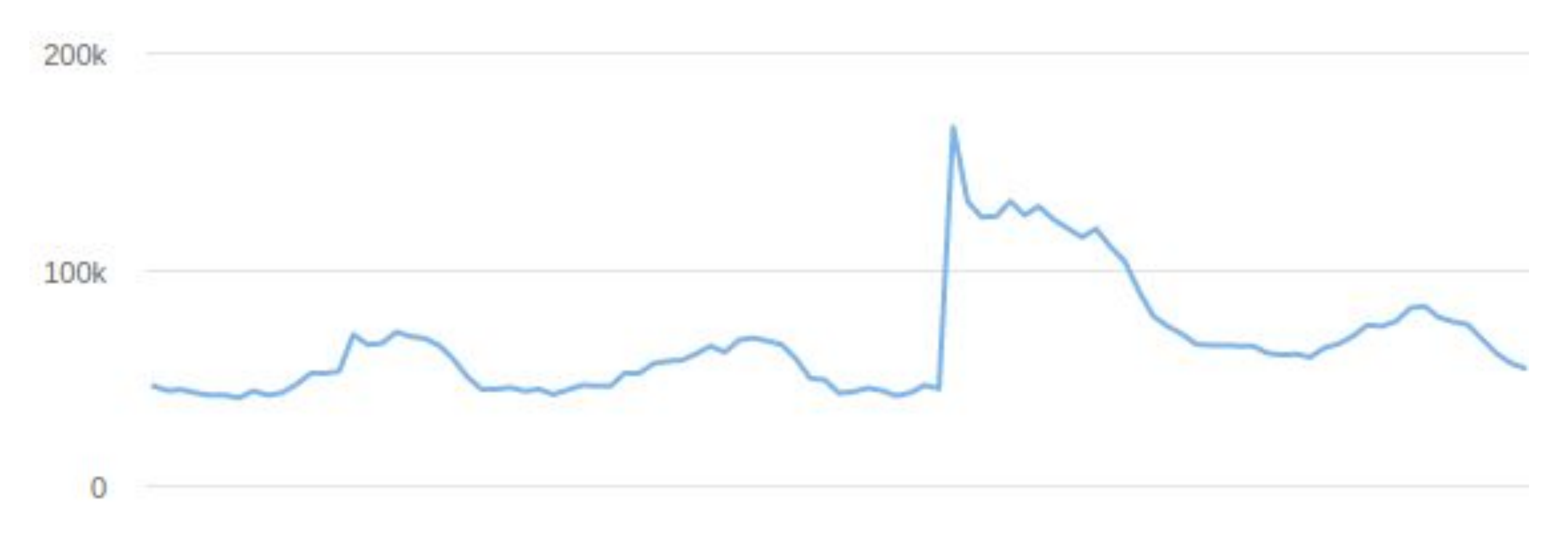
Предсказания для каждого горизонта соответствуют последнему наблюдаемому значению

Данная модель не должна использоваться для предсказаний, только для сравнения с другими

```
SELECT id, value AS prediction
FROM values_and_predictions
WHERE ts = toDateTime('#ts#') - #frequency#
```

Предсказания:

Наивная



Предсказания:

Наивная



Предсказания:

Скользящее среднее

$$SMA_t = \frac{1}{n}\sum_{i=0}^{n-1} p_{t-i}$$

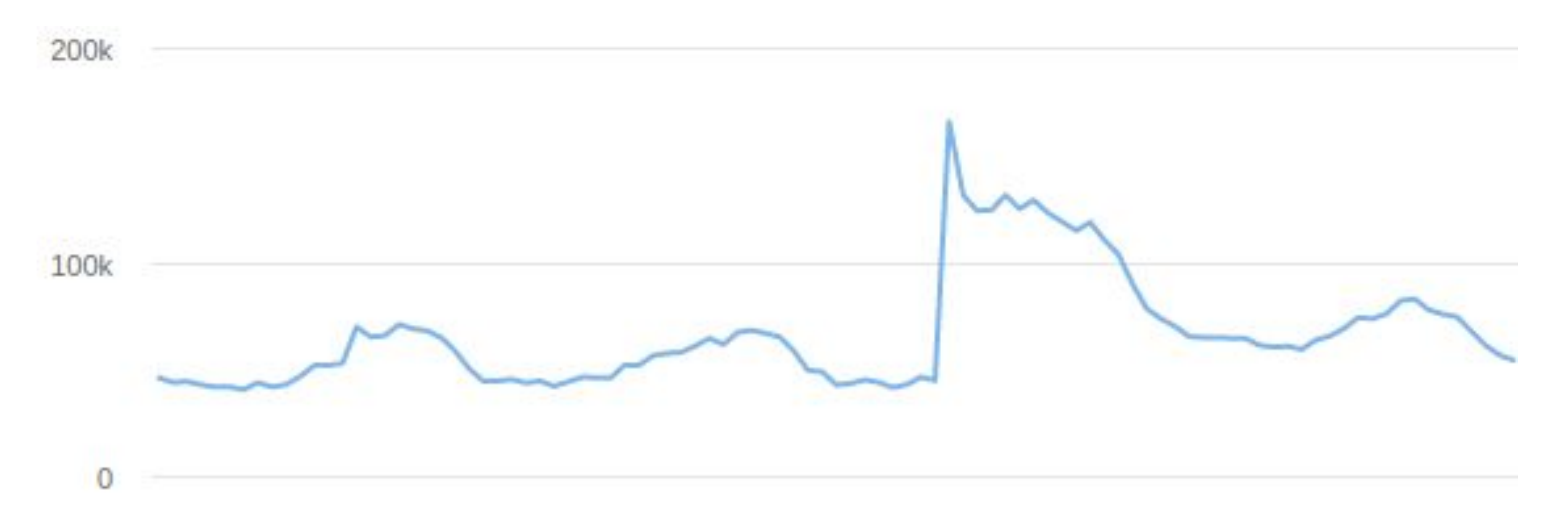
Предсказания численно равны среднему арифметическому значений исходной функции за установленный период

Отлично подходит под данные с выраженной сезональностью

```
select id, if(count() = length([#periods#]), AVG(value), nan) as prediction
from values_and_predictions
where time_period IN (#periods#)
group by id
```

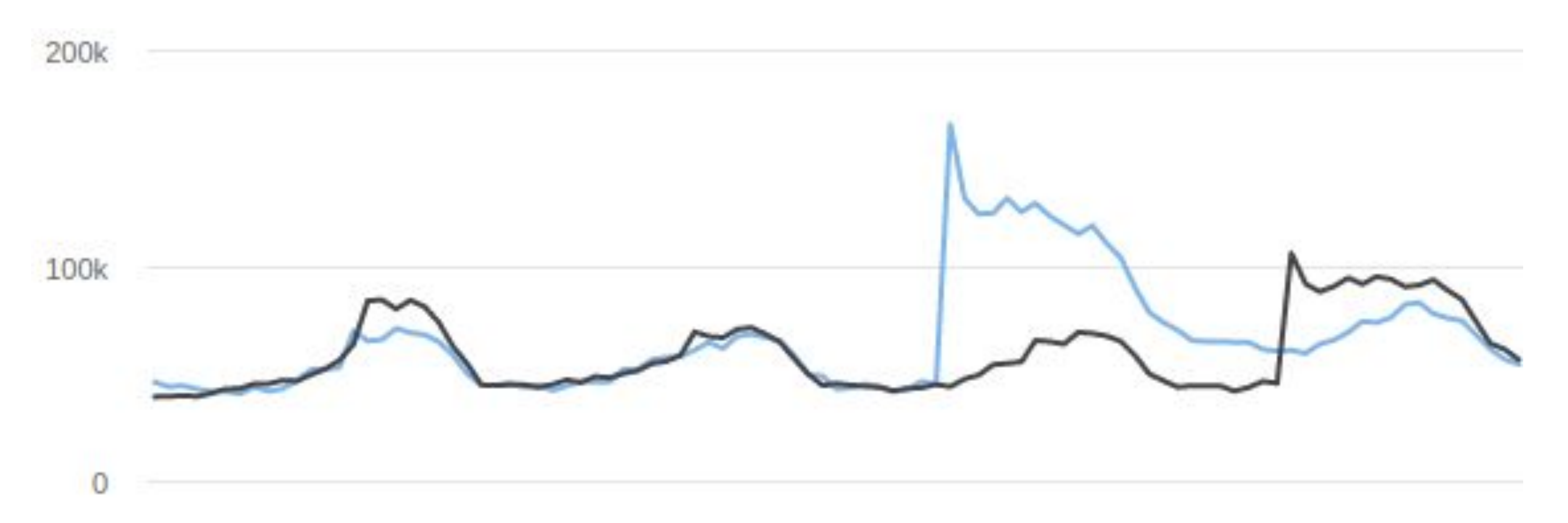
Предсказания:

Скользящее среднее



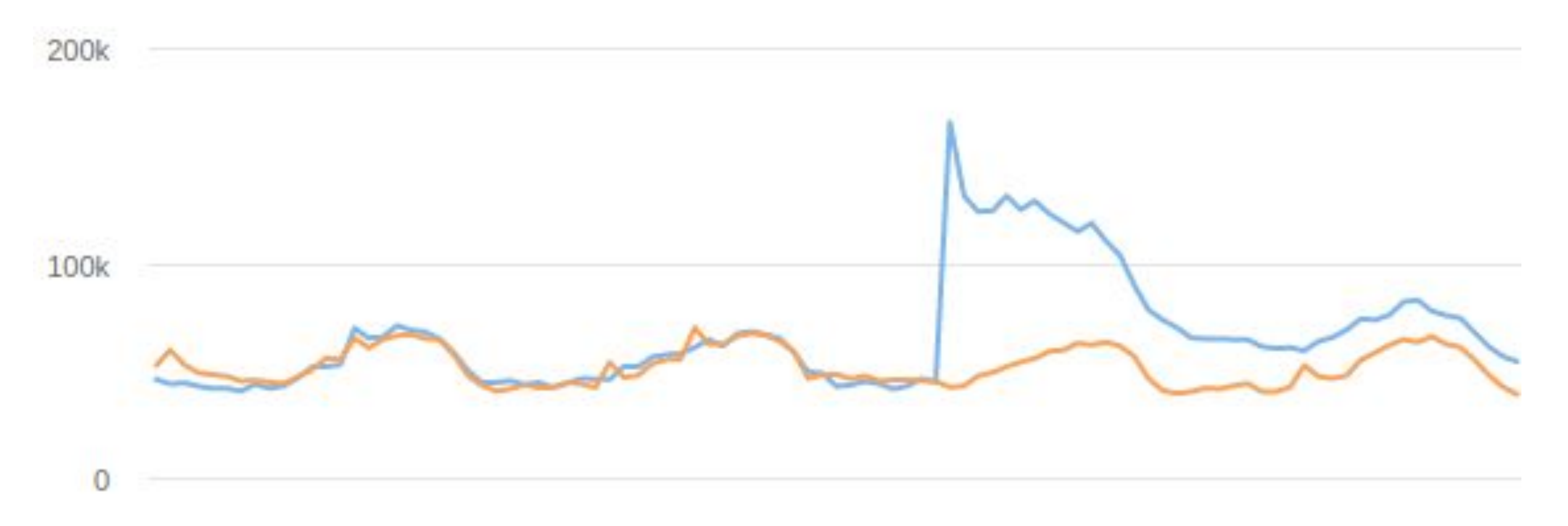
Предсказания:

Скользящее среднее



Предсказания:

Скользящее среднее



Предсказания:

Линейная регрессия

$$y = f(x, b) + \varepsilon,\ E(\varepsilon) = 0$$

$$f(x, b) = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_k x_k$$

Линейная регрессия — регрессионная модель зависимости одной переменной y от другой или нескольких других переменных (факторов, регрессоров, независимых переменных) x с линейной функцией зависимости

Отлично предсказывает данные, обладающие выраженным трендом

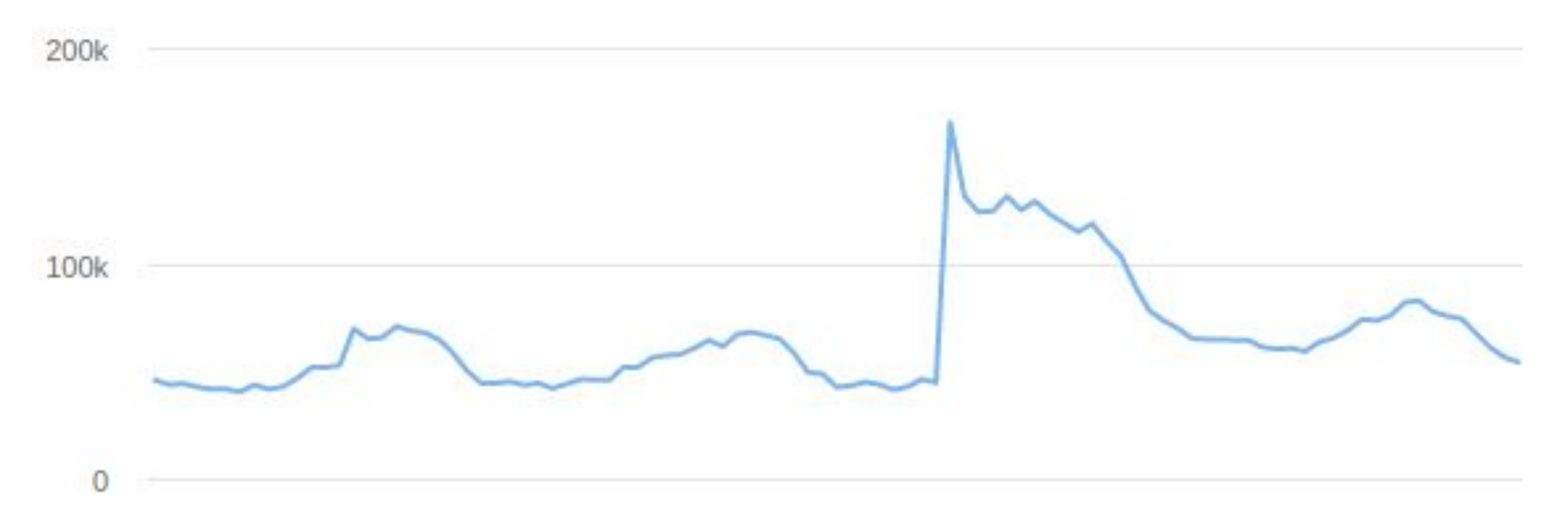
Предсказания:

Линейная регрессия

```
with
   #frequency# as frequency,
   toDateTime('#time period#') - frequency as upper_bound,
   upper bound - #seconds back# as lower bound,
   (time period - lower bound) / frequencyas idx,
   count() as num probes,
   sum(idx) as sum idx,
   max(idx) as max idx,
   max idx + 1 as result idx,
   sum(value) as sum values,
   sum(idx * value) as sum value_idx,
   sum(idx * idx) as sum idx sq,
   (sum values * sum idx sq  - sum idx * sum value idx) / (num probes * sum idx sq - sum idx * sum idx) as a,
   (num_probes * sum_value_idx - sum_idx * sum_values) / (num_probes * sum_idx_sq - sum_idx * sum_idx) as b
select
   id,
   if(num probes = #num probes#, a + b * result_idx,nan) as forecast
from values and predictions
where time period in ('#periods#')
group by id
```

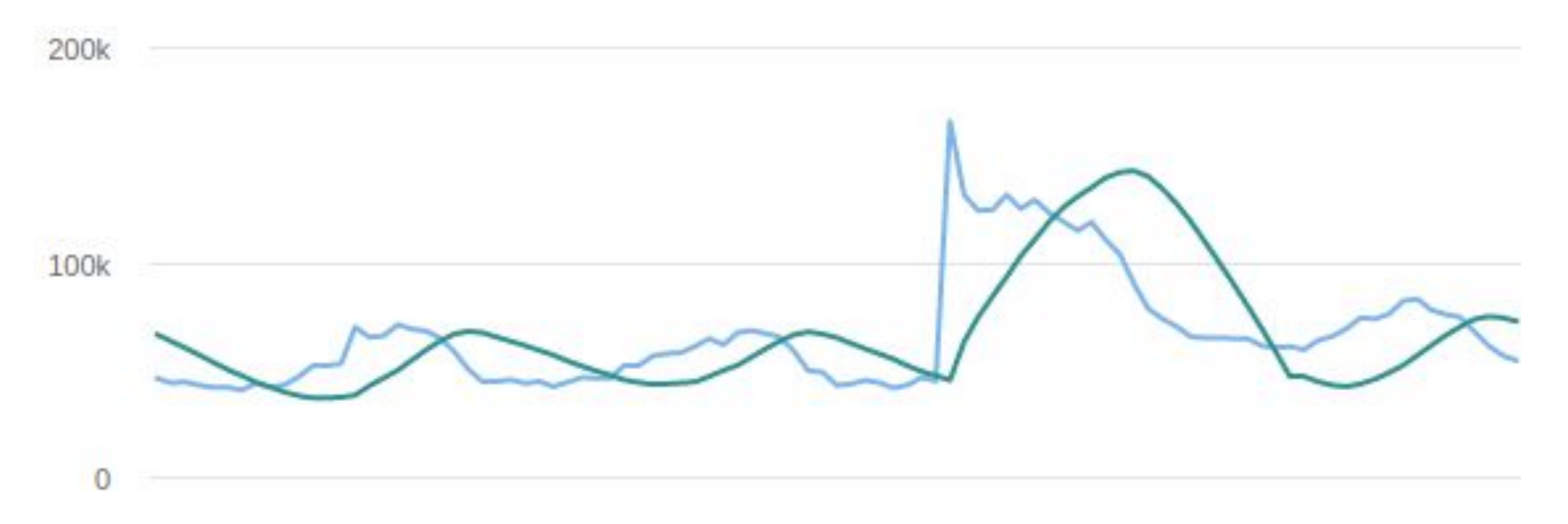
Предсказания:

Линейная регрессия



Предсказания:

Линейная регрессия



Какие модели еще можно использовать

- Экспоненциальное сглаживание
- ARIMA, SARIMA
- GARCH
- TBATS
- Prophet
- NNETAR
- LSTM

Доверительный интервал

Методы определения

Mean Square Error — средний квадрат ошибки определения какой-либо величины, квадратный корень из MSE есть среднеквадратическое отклонение определяемой величины от её математического ожидания.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Методы определения

Выборочная дисперсия — это среднее арифметическое квадратов отклонений всех вариант выборки от её средней

$$\sigma^2 = \frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2}{n}$$

Методы определения

Распределение Стьюдента используется, например, в t-критерии Стьюдента для оценки статистической значимости разности двух выборочных средних при построении доверительного интервала для математического ожидания

Позволяет получить доверительный интервал с нужной точностью при малых объемах выборки

Коэффициенты (critical values) не зависят от данных и можно использовать табличные значения

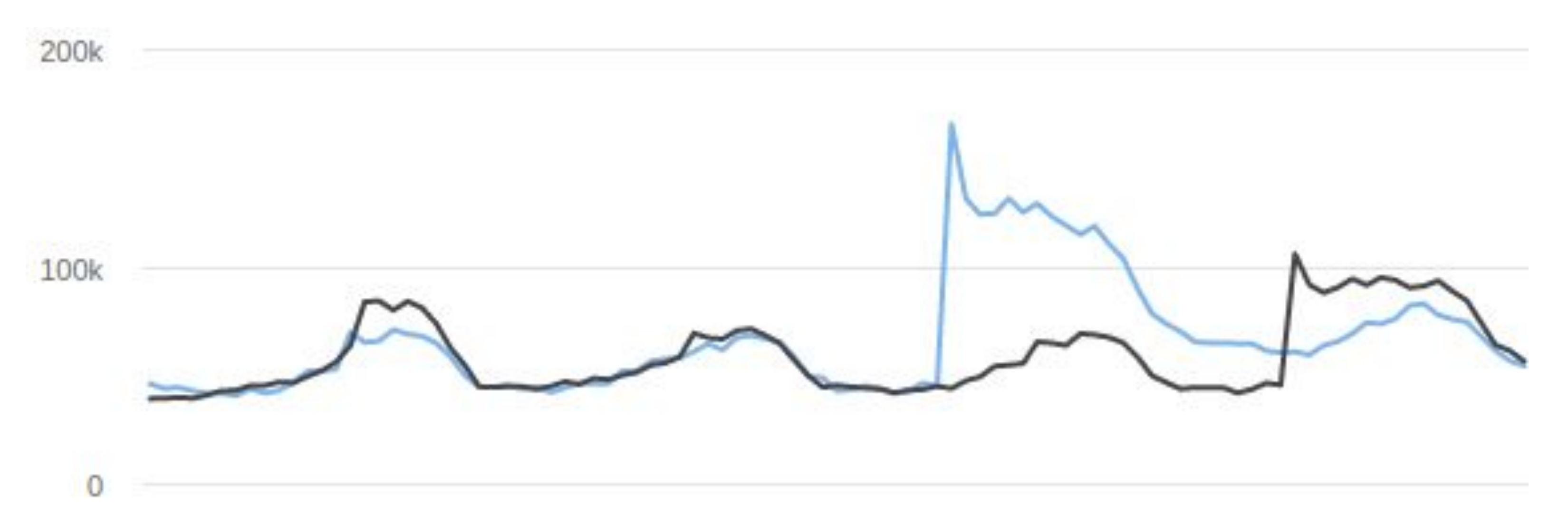
Доверительный интервал:

Запрос

```
INSERT INTO values_and_predictions
(id, ts, threshold_#HASH#)
WITH
   prediction_#HASH# AS expected,
   expected - value AS error,
   sum(pow(error, 2)) AS sum_squared_errors,
   varPop(expected) AS predictions_variance,
   count() AS sample_size,
   30 AS max_sample_size,
   if(sample_size < max_sample_size, sample_size, max_sample_size)AS number_of_degrees,
   [#t_critical_values#][number_of_degrees]AS t_critical_value
SELECT
   id,
   toDateTime('#time_period#'),
   t_critical_value * sqrt(predictions_variance + sum_squared_errors) /sqrt(number_of_degrees) AS threshold
FROM values_and_predictions
WHERE ts BETWEEN toDateTime('#start_time_period#') AND computation_period
GROUP BY id
```

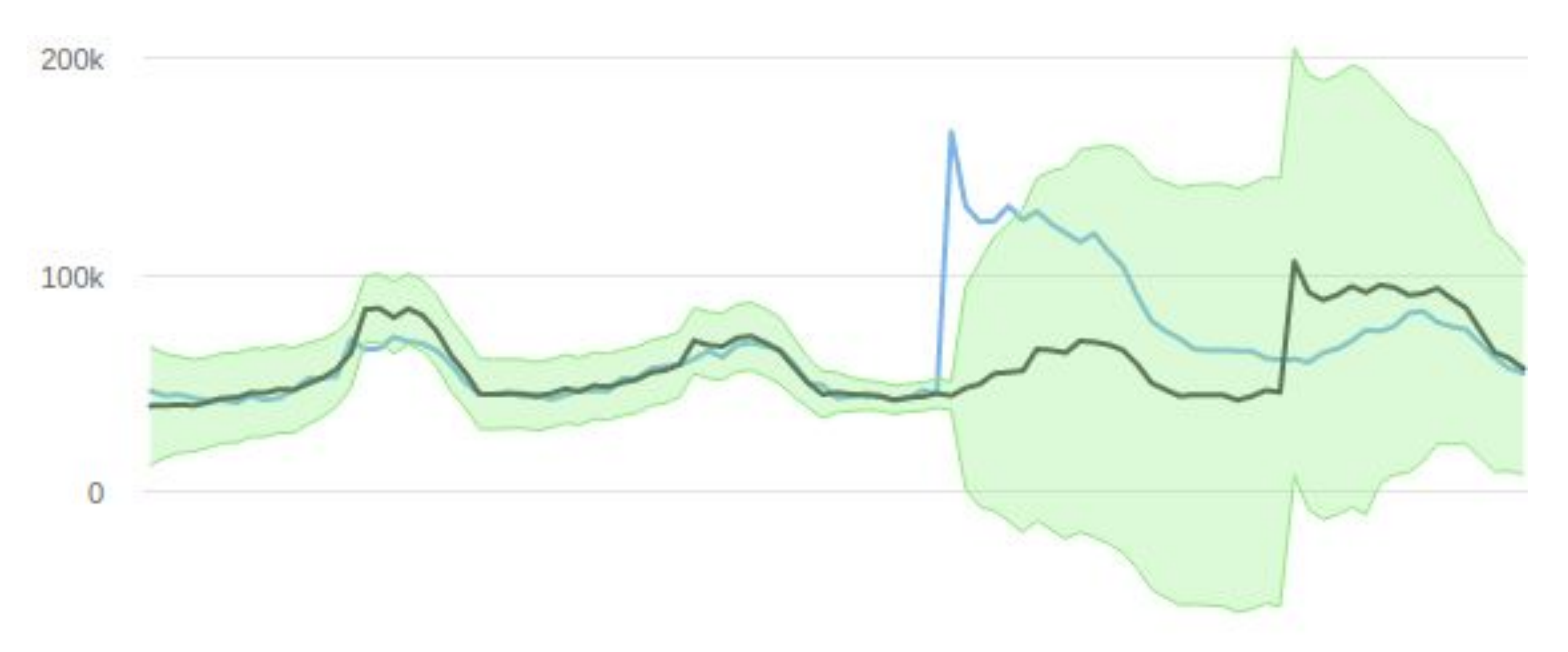
Доверительный интервал:

Пример



Доверительный интервал:

Пример



Доверительный интервал:

Результаты

- Умеем работать с шумными метриками
- Реагируем на аномалии
- Работаем с малым объемом данных

Лучшая модель

Лучшая модель:

Что мы уже имеем?

- Данные подготовлены
- Предсказания сделаны
- Доверительные интервалы посчитаны

- Для каждой точки на каждой линии мы имеем:
  - 1 реальное значение
  - N предсказаний
  - N доверительных интервалов

Как выбрать лучшую модель?

Решение: лучшая — это та, что ошибается меньше всего

Ограничения:

- Анализируем четко определенное количество точек в прошлом
- Все ожидаемые модели должны быть подсчитаны для всех точек

Лучшая модель:

Ошибка модели предсказаний

WMAPE (иногда пишется wMAPE) означает средневзвешенную абсолютную ошибку в процентах

$$\mathrm{WMAPE} = \frac{\sum_{t=1}^{n} |A_t - F_t|}{\sum_{t=1}^{n} |A_t|}$$

Считаем ошибку для всех моделей на всех точках

Выбираем модель с наименьшей ошибкой

Лучшая модель:

Запрос

```
WITH
   sum(value) AS sum_value,
   if(sum_value = 0, 1e-10, sum_value) AS sum_value_safe
SELECT
   id,
   arraySort((x, y) -> y, [#model_names#], [#model_wmape_expressions#])[1] as best_model
FROM values_and_predictions
GLOBAL INNER JOIN (
   select id
   from values_and_predictions
   where
       ts = toDateTime('#time_period#') - INTERVAL #retention# DAY
       and bitAnd(#models_num#, mask) = #models_num#
) USING (id)
WHERE
   ts BETWEEN (toDateTime('#time_period#') - INTERVAL 7 DAY) AND '#time_period#'
   AND bitAnd(#models_num#, mask) = #models_num#
GROUP BY id
HAVING count() >= #min_num_forecasts#
```

Лучшая модель:

Запрос

Для каждой модели заранее подготавливаем выражение подсчета ошибки и подставляем в

```
model_wmape_expressions
```

```
sum(abs(value - prediction_#HASH#)) /sum_value_safe
```

Лучшая модель:

Запрос

Лимитируем выборку по нужным моделям в первой точке

```
GLOBAL INNER JOIN (
   select id
   from values_and_predictions
   where
       time_period = toDateTime('#time_period#') - INTERVAL 7 DAY
       and bitAnd(#models_num#, mask) = #models_num#
) USING (id)
```

Лучшая модель:

Запрос

Берем лучшую модель только для тех графиков, где количество точек достаточно

```
HAVING count() >= #min_num_forecasts#
```

Итого

Мы имеем:

- Много графиков с предсказаниями и доверительными интервалами
- Знание о лучших моделях

Сохраняем результаты

Что у нас есть?

- Данные подготовлены
- Предсказания сделаны
- Доверительные интервалы посчитаны
- Лучшая модель определена

Заполняем историю

Забираем данные из **values_and_predictions**

Забираем метаданные из **dimensions**

Собираем дополнительную аналитику, если надо

Вставляем все в **history**

Что делать, когда много аномалий?

Много аномалий:

Получение аномалий

```
select * from history
where
   abs(predicted_value - actual_value) > threshold
   and ts = toDateTime('#time_period#')
   and #filter_expression#
order by actual_value
```

Как сортировать?

1. Просто
   Больше цифра — важнее аномалия

2. Хитро
   Больше отклонение — важнее аномалия

Много аномалий:

Как фильтруем

1. По параметрам графика
2. По значению графика
3. По поведению графика
4. По времени суток
5. По продолжительности аномалии

Много аномалий:

Как фильтруем — ключ к победе

Сложная логика фильтров

Комбинация поведенческих фильтров с параметрическими

Результат: возможность точно указать, какие аномалии нужно показать

Итоги

Итоги:

Что мы узнали?

Построить систему поиска аномалий можно

Можно обрабатывать гигантские объемы данных

Пользоваться решением могут все

Спасибо!
Вопросы?